Advances in Computational Mathematics manuscript No.
(will be inserted by the editor)

Evaluation of generalized Mittag–Leffler functions on the real line

Roberto Garrappa · Marina Popolizio

Received: date / Accepted: date

Abstract This paper addresses the problem of the numerical computation of generalized Mittag–Leffler functions with two parameters, with applications in fractional calculus. The inversion of their Laplace transform is an effective tool in this direction; however, the choice of the integration contour is crucial. Here parabolic contours are investigated and combined with quadrature rules for the numerical integration. An in–depth error analysis is carried out to select suitable contour's parameters, depending on the parameters of the Mittag–Leffler function, in order to achieve any fixed accuracy. We present numerical experiments to validate theoretical results and some computational issues are discussed.

Keywords Mittag–Leffler function · Laplace transform · fractional calculus · contour integral

Mathematics Subject Classification (2000) 33E12 · 44A10 · 26A33 · 65D30

1 Introduction

The Mittag–Leffler (ML) function was introduced, at the beginning of the last century, by the Swedish mathematician Magnus Gösta Mittag–Leffler in the context of summing divergent series. The formal definition

$$E_{\alpha}(z) = \sum_{k=0}^{\infty} \frac{z^k}{\Gamma(\alpha k + 1)} \tag{1}$$

Roberto Garrappa
Università degli Studi di Bari "Aldo Moro", Dipartimento di Matematica Via E. Orabona n.4, 70125 Bari, Italy, E-mail: garrappa@dm.uniba.it

Marina Popolizio
Università del Salento, Dipartimento di Matematica e Fisica "Ennio De Giorgi" Via per Arnesano, 73100 Lecce, Italy, E-mail: marina.popolizio@unisalento.it

holds for complex z's and α's with $\Re(\alpha) > 0$. Shortly after its introduction this definition was broadened with a second complex parameter β, $\Re(\beta) > 0$, according to the expression

$$E_{\alpha,\beta}(z) = \sum_{k=0}^{\infty} \frac{z^k}{\Gamma(\alpha k + \beta)}. \qquad (2)$$

However, essentially in the last two decades ML functions have come into prominence thanks to the vast potential of their applications in modeling systems from physics, biology, engineering and many other fields.

A fundamental application area is the fractional calculus, as widely discussed in some cornerstone books [4,15,19,21,26], since ML functions play a key role in the solution of fractional order problems (e.g., see [3,8,32]).

Recent developments in the solution of fractional differential equations (FDEs) involve a generalization of Mittag–Leffler functions, namely

$$e_{\alpha,\beta}(t;\lambda) = t^{\beta-1} E_{\alpha,\beta}(-t^{\alpha}\lambda), \qquad (3)$$

where t is the independent variable, while λ, α and β are fixed (usually real) parameters. We will refer to $e_{\alpha,\beta}$ as the *generalized Mittag–Leffler function*.

Several and important results have been given for the ML functions (2) and recently also for (3) (e.g., see [12,18,19,22,26,28]), although important aspects still deserve a deep comprehension. In particular, the case $\alpha = \beta = 1$ corresponds to the exponential function and the case $e_{\alpha,1}(t;1)$ is discussed in–depth in [11,20].

In several applications the evaluation of the generalized ML function $e_{\alpha,\beta}$ is requested for matrix arguments. This topic has been extensively discussed in recent works and we just cite, among the others, [7,5,23,24,35]. Very often the attention is in the computation, in a fast way, of the action of this matrix function on a vector. An example is given by the following system of linear FDEs

$$y^{(\alpha)}(t) + Ay(t) = b, \quad y(0) = y_0,$$

where $y(t)$ is a vector–valued function, $y^{(\alpha)}(t)$ its fractional derivative of order $0 < \alpha < 1$ according to the Caputo's definition and A, b and y_0 respectively matrix and vectors of compatible size; the solution of this problem can be expressed as $y(t) = e_{\alpha,1}(t;A)y_0 + e_{\alpha,\alpha+1}(t;A)b$ [4,19,26]. We note that in several and important applications the size of the system is very large.

In the case in which the input b is not constant but varies with respect to the time t or it is a function of the state $y(t)$, the application of *Generalized Exponential Integrators* (GETD) [5] allows to reduce the above problem to the evaluation of several vectors of the form $e_{\alpha,\beta}(t;A)v$ for different values of t and v.

The increasing interest in ML functions and their wide application give rise to the need of effective strategies for their numerical computation [10,13, 29,27]. Thus the analysis of efficient and accurate numerical methods for (3), with scalar or matrix arguments, is actually a compelling subject.

In this paper we address this topic by considering a technique based on the numerical inversion of the Laplace transform combined with standard quadrature rules on a parabolic contour. This approach has been extensively studied in [35] and is extremely elegant and simple; there are very few parameters to select and they can be easily and automatically tuned to meet a given accuracy.

Anyway, as we will see later, the procedure described in [35] fails to achieve good results when β is large enough as compared to α. The fundamental result of this paper is the description of the explicit dependence of the error on both parameters α and β; grounding on this result, we are able to select appropriate parabolic contours to accurately evaluate the generalized ML function (3).

In this paper we focus on the case in which the function argument is a positive real value or a matrix with the spectrum on the positive real axis; we outline that these cases have several applications in real–life models.

The structure of the paper is the following: in Section 2 some important properties of the Laplace transform of $e_{\alpha,\beta}$ are discussed and the main results concerning its inversion by integration on parabolic contours are reviewed. Section 3 is devoted to study the error estimation in the numerical inversion of the Laplace transform of $e_{\alpha,\beta}$ and the problem of finding the suitable contour to reach a prescribed tolerance is addressed. Some numerical experiments are presented in Section 4 to validate theoretical results, both for scalar and matrix arguments. Finally, in Section 5 we discuss some issues related to the computational efficiency of the proposed method.

2 On the inversion of the Laplace transform of $e_{\alpha,\beta}(t;\lambda)$

Whilst most of the analytical properties of special functions, such as the generalized ML function investigated in this paper, have been widely studied, their numerical evaluation is a still challenging task. We refer for instance to one of the books wholly devoted to this subject [9]. A powerful class of methods for performing this task is based on the numerical inversion of the Laplace transform.

It is a well–known result (e.g., see [19,26]) that the Laplace transform of $e_{\alpha,\beta}(t;\lambda)$ is

$$F_{\alpha,\beta}(s;\lambda) = \frac{s^{\alpha-\beta}}{s^{\alpha}+\lambda}, \quad \Re(s) > |\lambda|^{1/\alpha};$$

due to the presence of non integer powers, $F_{\alpha,\beta}(s;\lambda)$ is a multivalued function and to prevent related problems we define a branch cut along the negative real semiaxis.

An effective way to compute $e_{\alpha,\beta}(t;\lambda)$ is therefore approximating the inversion Laplace transform formula

$$e_{\alpha,\beta}(t;\lambda) = \frac{1}{2\pi i}\int_{\mathcal{C}} e^{st} F_{\alpha,\beta}(s;\lambda) ds; \tag{4}$$

the contour $\mathcal{C}$ is a suitable deformation of the Bromwich line which should be located as much as possible to the left to keep the factor e^{st} at a small size but, at the same time, $\mathcal{C}$ should not be too close to the singularities of $F_{\alpha,\beta}$ to avoid the blow up of the integrand function.

A fundamental task is therefore the identification of a region in the complex plane in which $F_{\alpha,\beta}(s;\lambda)$ is analytic. We first note that, unless $\beta \le \alpha$, a first singularity of $F_{\alpha,\beta}(s;\lambda)$ is located on the brunch point $s=0$. Further feasible singularities are of the form

$$s_{\alpha,\lambda}^{\{k\}} = \lambda^{\frac{1}{\alpha}} e^{i\frac{\pi+2k\pi}{\alpha}}, \quad k \in \mathbb{Z}$$

since we assumed λ to be on the positive real semi–axis.

With the above choice of the brunch cut, when $0 < \alpha < 1$ none of the above possibility has the argument in $(-\pi, \pi]$ and hence $F_{\alpha,\beta}(s;\lambda)$ has only the singularity at the origin. When $1 < \alpha < 2$ we have to take into account the two singularities at

$$s_{\alpha,\lambda}^{\pm} = \lambda^{\frac{1}{\alpha}} e^{\pm i\frac{\pi}{\alpha}}$$

and, since $\frac{\pi}{2} < \frac{\pi}{\alpha} < \pi$, these singularities have negative real part. We summarize the above discussion in the following result and we omit to treat the case $\alpha > 2$ (which presents more singularities and raises additional difficulties) since it is usually of less interest in applications.

Proposition 1 *Let $0 < \alpha < 2$ and $\lambda > 0$. There exists a sector*

$$\Sigma_{\delta,\sigma} = \bigl\{ s \in \mathbb{C} \, , \, |\arg(s-\sigma)| < \pi - \delta \bigr\},$$

where $\sigma > 0$ and $0 \le \delta \le \max\{0, \pi(\alpha-1)/\alpha\}$, such that $F_{\alpha,\beta}(s;\lambda)$ is analytic in $\Sigma_{\delta,\sigma}$.

The numerical evaluation of (4) has been addressed in [6] where, by following ideas in [30], the exponential function was replaced by a rational approximation. This strategy has the main advantage that it makes possible to achieve high accuracy with very few nodes. Anyway it has two non negligible weak points: the first is that the error analysis could be not always sharp since it is based merely on a conjecture (even if validated by several experimental results); secondly, since this replacement is equivalent to the integration on a certain contour, it is not applicable to the case in which the integrand presents some singularities since we are not sure that the singularities lie on the left of the contour.

The integration on a well defined contour, such as a parabola or hyperbola, has been deeply discussed, for instance, in [16,17,33–35]; it usually involves more evaluations of the Laplace transform to be inverted but it allows to tune in a fine way the parameters in order to keep singularities at a safe distance from the contour and, moreover, provides an analytic error estimation.

Because of the presence of some singularities, the opinion of the authors, based also on numerical tests shown in the following, is that the inversion of the function $F_{\alpha,\beta}$ should be preferably done by methods that, unlike the one in [30], leave some freedom in moving the contour in a proper way. This aspect is the main motivation of the analysis in this paper, which actually aims to select contours depending on the problem data, in such a way to optimally solve it.

Given a contour parametrized by a complex function $z(u)$, $-\infty < u < \infty$, we can write the integral (4) as

$$e_{\alpha,\beta}(t;\lambda) = \frac{1}{2\pi i}\int_{-\infty}^{\infty} e^{z(u)t}\frac{z(u)^{\alpha-\beta}}{z(u)^{\alpha}+\lambda}z'(u)du\,. \tag{5}$$

The approximation by means of the trapezoidal rule on a uniform mesh–grid $u_k = kh$ with step size h gives the approximation

$$e_{\alpha,\beta}^{[h,N]}(t;\lambda) = \frac{h}{2\pi i}\sum_{k=-N}^{N} e^{z(u_k)t}\frac{z(u_k)^{\alpha-\beta}}{z(u_k)^{\alpha}+\lambda}z'(u_k). \tag{6}$$

The analysis of the error $e_{\alpha,\beta}(t;\lambda) - e_{\alpha,\beta}^{[h,N]}(t;\lambda)$ can be made by following the approach in [35]. More generally, given the integral

$$I = \int_{-\infty}^{\infty} g(u)du,$$

its infinite and finite trapezoidal approximations are

$$I_h = h\sum_{k=-\infty}^{\infty} g(kh), \quad I_{h,N} = h\sum_{k=-N}^{N} g(kh);$$

the error $I - I_{h,N}$ is given by the discretization error $DE = I - I_h$ and the truncation error $TE = I_h - I_{h,N}$. For the truncation error it is sufficient to assume that $g(u)$ decays rapidly as $u \to \pm\infty$ in order to approximate TE by the last term retained in the summation, i.e. $TE = \mathcal{O}\bigl(|g(hN)|\bigr)$, $N \to \infty$; on the contrary, for the discretization error a more involved analysis is necessary. The main tool is the following theorem which we report here for completeness.

Theorem 1 ([35]) *Let $w = u+iv$, with u and v real. Suppose $g(w)$ is analytic in the strip $-d < v < c$, for some $c > 0$, $d > 0$, with $g(w) \to 0$ uniformly as $|w| \to \infty$ in that strip. Suppose further that for some $M_+ > 0$, $M_- > 0$ the function $g(w)$ satisfies*

$$\int_{-\infty}^{\infty}\bigl|g(u+ir)\bigr|du \le M_+, \quad \int_{-\infty}^{\infty}\bigl|g(u-is)\bigr|du \le M_-$$

for all $0 < r < c$, $0 < s < d$. Then

$$\bigl|I - I_h\bigr| \le DE_+ + DE_-,$$

where

$$DE_+ = \frac{M_+}{e^{2\pi c/h}-1}, \quad DE_- = \frac{M_-}{e^{2\pi d/h}-1}.$$

In [35] parabolic and hyperbolic contours were analyzed and compared for general functions; for the ML function (1), i.e. for the special case $\beta = 1$, the attention was restricted to the parabolic contour of the form

$$\mathcal{C} \, : \, z(u) = \mu\big(iu + 1\big)^2, \quad -\infty < u < \infty. \tag{7}$$

Throughout the paper we will assume $\mu > 0$ to be sure that the parabola encloses the real negative axis.

For the symmetry of the contour, the coefficients in the sum (6) can be rearranged in such a way that only half of them are actually computed and complex arithmetic is halved.

One of the most innovative aspects of the analysis carried out in [35] is the use of a non–symmetric strip of analyticity. Whilst this is not particularly revealing from a theoretical point of view, it makes the difference for practical computation since it allows to select parameters in an optimal way. Indeed, based on the results in Theorem 1, in [35] it was given the balance of the components DE_+, DE_- and TE of the error for the ML function (1) when using the parabola (7); suitable parameters are thus obtained as

$$h = \frac{3}{N}, \quad \mu = \frac{\pi N}{12t}, \tag{8}$$

where N is chosen such that the error $E_N = \mathcal{O}(e^{-2\pi N/3})$ is proportional to a given tolerance $\varepsilon > 0$.

Unfortunately, the parameters found in [35] do not always work properly for the generalized ML function (3). In Figure 1 we show, for $\lambda = 2.5$ and $t = 1$, the errors obtained by selecting h and μ according to (8) with target tolerance of $\varepsilon = 10^{-8}$ and $\varepsilon = 10^{-11}$. The evaluations have been performed for $\beta \in \{\alpha, 1, \alpha+1, \alpha+2, \alpha+3, \alpha+4\}$ with $\alpha = 0.2, 0.5, 0.8$. The straight dotted line represents the target tolerance which is reached only for small values of β whilst for higher values of β errors greater than the expected ones are obtained.

We observe that the distance between the actual error and the target tolerance increases as β increases. Our opinion is that, in the case of the ML function, some terms with large size have been neglected in the asymptotic representations of the errors DE_+, DE_- or TE and some of these terms could be in some way related to α and β. The main aim of this paper is thus to bring to light these terms and devise a more accurate error model for the computation of the ML function.

3 Contours tuning for the generalized ML function

The main reason for the poor performance of the procedure described in Section 2 when applied to the generalized ML function (3) is that the constant M_+ in Theorem 1 can assume extremely large values when β is larger than α. Indeed, as $r \to 1$ the parabola described by $z(u + ir)$ approaches the negative real axis and, since the singularity at the origin, the value of $g(u + ir)$ blows up.

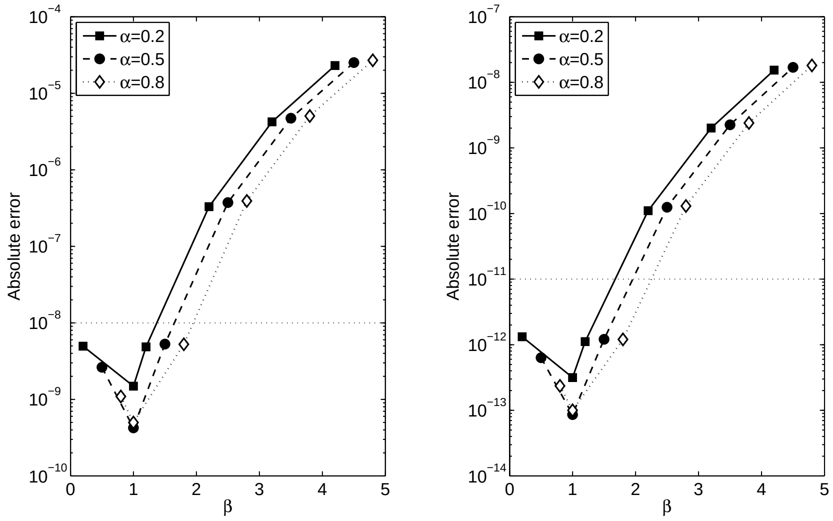

Fig. 1 Errors for the function $e_{\alpha,\beta}(t;\lambda)$ by using classical parameters (8) with target tolerance of $\varepsilon = 10^{-8}$ (left) and $\varepsilon = 10^{-11}$ (right).

To prevent this from happening, r should not be allowed to get too close to 1 and should be constrained by a threshold $c < 1$. In this way the balancing of the different components of the error would select optimal parameters h and μ leading to the global error $E_N = \mathcal{O}\big(e^{-2\pi Nc/\sqrt{1+4c(1+c)}}\big)$. Thus, any tolerance $\varepsilon > 0$ is reached by using a number of nodes given by

$$N = -\frac{\sqrt{1+4c(1+c)}}{2\pi c}\log\varepsilon. \tag{9}$$

The threshold c should be selected small enough to guarantee M_+ be not large when compared to $e^{-2\pi c/h}$. However, from (9) we infer a direct dependence of the number of nodes N on c: if a too much small value of c is used, a huge number N of nodes is required, thus impairing the method efficiency. For this reason it is fundamental to detect an optimal value for the width c of the strip of analyticity in which M_+ is evaluated.

To achieve this goal, we need a deeper analysis of the component DE_+ in the error estimate.

3.1 The case $0 < \alpha < 1$

Let us start by first introducing some results which represent a preliminary to our analysis. In [31] Tricomi introduced the Ψ–function as the second solution of the confluent hypergeometric differential equation (also called Kummer's equation)

$$z\frac{d^2y}{dz^2} + (b-z)\frac{dy}{dz} - ay = 0.$$

The Ψ–function, which very often is referred to as the confluent hypergeometric function of the second kind, possesses for any a and z with positive real part the following integral representation

$$\Psi(a,b,z) = \frac{1}{\Gamma(a)} \int_0^{\infty} e^{-zs} s^{a-1} (1+s)^{b-a-1} ds. \tag{10}$$

The use of the Ψ–function allows to describe in a functional analytic way the dependence of M_+ from r in the formula for the discretization error DE_+.

Lemma 1 *Let $a, p > 0$ and $\gamma \in \mathbb{R}$. Then*

$$\int_{-\infty}^{\infty} e^{-pu^2} \left(a+u^2\right)^{\gamma} du = \sqrt{\pi} a^{\gamma+\frac{1}{2}} \Psi\bigl(\tfrac{1}{2}, \gamma+\tfrac{3}{2}, ap\bigr).$$

Proof First split the integral into the two subintervals $(-\infty, 0]$ and $[0, \infty)$ and, after the change of variable $s = u^2/a$, observe that

$$\int_{-\infty}^{\infty} e^{-pu^2} \left(a+u^2\right)^{\gamma} du = a^{\gamma+\frac{1}{2}} \int_0^{\infty} s^{-\frac{1}{2}} e^{-aps} \bigl(1+s\bigr)^{\gamma} ds.$$

By using the integral representation (10) it is immediate to see that

$$\int_{-\infty}^{\infty} e^{-pu^2} \left(a+u^2\right)^{\gamma} du = a^{\gamma+\frac{1}{2}} \Gamma(\tfrac{1}{2}) \Psi\bigl(\tfrac{1}{2}, \gamma+\tfrac{3}{2}, ap\bigr)$$

and the proof now follows after using the equivalence $\Gamma(\frac{1}{2}) = \sqrt{\pi}$ (e.g., see [25]). □

Theorem 2 *Let $0 < r < 1$ and let g denote the integrand of (5) for the contour z defined by (7). Then*

$$\int_{-\infty}^{\infty} \bigl|g(u+ir)\bigr| du \leq M(1-r)^{2(\alpha-\beta+1)} \Psi\bigl(\tfrac{1}{2}, \alpha-\beta+2, \mu t(1-r)^2\bigr),$$

where M is a positive constant which does not depend on r.

Proof By evaluating the parabola on $w = u + ir$ it is immediate to see that

$$\bigl|z(u+ir)^{\alpha-\beta}\bigr| = \mu^{\alpha-\beta} \bigl((1-r)^2+u^2\bigr)^{\alpha-\beta}, \ \bigl|z'(u+ir)\bigr| = 2\mu\sqrt{(1-r)^2+u^2}$$

and, since $\bigl|e^{z(u+ir)t}\bigr| = e^{\mu(1-r)^2 t} e^{-\mu u^2 t}$, we easily obtain

$$\bigl|g(u+ir)\bigr| = 2e^{\mu(1-r)^2 t} \mu^{\alpha-\beta+1} e^{-\mu u^2 t} \bigl((1-r)^2+u^2\bigr)^{\alpha-\beta+\frac{1}{2}} \frac{1}{\bigl|z(u+ir)^{\alpha}+\lambda\bigr|}.$$

Moreover, since for $0 < \alpha < 1$ it is $z(u+ir)^{\alpha} + \lambda \neq 0$, we can assume there exists a positive value $\tilde{M}$ such that

$$\frac{1}{\bigl|z(u+ir)^{\alpha}+\lambda\bigr|} \leq \tilde{M}$$

and, since $e^{\mu(1-r)^2 t} \le e^{\mu t}$, for $M_1 = 2e^{\mu t}\mu^{\alpha-\beta+1}\tilde{M}$ we have

$$\int_{-\infty}^{\infty} \bigl|g(u+ir)\bigr| du \le M_1 \int_{-\infty}^{\infty} e^{-\mu u^2 t}\bigl((1-r)^2+u^2\bigr)^{\alpha-\beta+\frac{1}{2}} du.$$

We can now apply Lemma 1 for $\gamma = \alpha - \beta + \frac{1}{2}$, $p = \mu t$ and $a = (1-r)^2$ in order to evaluate the above integral and hence conclude the proof with $M = \sqrt{\pi} M_1$. □

We are now able to study the role of the parameter c in the error term DE_+ by means of the following result.

Theorem 3 *Let* $0 < r \le c < 1$. *Then*

$$\int_{-\infty}^{\infty} \bigl|g(u+ir)\bigr| du \le \begin{cases} M_+^{(1)} & \textit{if } \beta < \alpha+1 \\ M_+^{(2)} \log(\mu t(1-c)^2) & \textit{if } \beta = \alpha+1 \\ M_+^{(3)} (1-c)^{2(\alpha-\beta+1)} & \textit{if } \beta > \alpha+1 \end{cases}$$

for some positive constant $M_+^{(i)}$, $i = 1, 2, 3$, *which depends on* α *and* β *but not on* c.

Proof Theorem 2 states that DE_+ can be expressed in terms of the function

$$f_{\alpha,\beta}(r;\mu t) \equiv (1-r)^{2(\alpha-\beta+1)} \Psi\bigl(\tfrac{1}{2}, \alpha-\beta+2, \mu t(1-r)^2\bigr);$$

we thus analyze its behavior as r approaches 1.

The following expansions for the Ψ–function, as $z \to 0$, turn out to be useful:

$$\Psi(a,b,z) = \begin{cases} \displaystyle\sum_{j=0}^{\infty} u_j^{(1)} z^j + z^{1-b} \sum_{j=0}^{\infty} v_j^{(1)} z^j & \text{if } b \notin \mathbb{Z} \\ \displaystyle\sum_{j=0}^{-b} u_j^{(2)} z^j + z^{1-b} \sum_{j=0}^{\infty} v_j^{(2)} z^j + \log(z) z^{1-b} \sum_{j=0}^{\infty} w_j^{(2)} z^j & \text{if } b \in \mathbb{Z}_0^- \\ \displaystyle\sum_{j=0}^{\infty} u_j^{(3)} z^j + \sum_{j=1}^{b-1} v_j^{(3)} z^{-j} + \log(z) \sum_{j=0}^{\infty} w_j^{(3)} z^j & \text{if } b \in \mathbb{Z}^+ \end{cases}$$

with $\mathbb{Z}_0^- \equiv \mathbb{Z}^- \cup \{0\}$ and $\{u_j^{(\ell)}\}_j$, $\{v_j^{(\ell)}\}_j$, $\{w_j^{(\ell)}\}_j$ some sequences of coefficients whose explicit formulation can be found in [36].

Indeed, when $\beta < \alpha+1$ only positive powers of $(1-r)$ appear in $f_{\alpha,\beta}(r;\mu t)$ and hence its limit as $r \to 1$ is the constant value $v_0^{(1)}(\mu t)^{\beta-\alpha-1}$, except for $\beta = \alpha$ in which case $f_{\alpha,\beta}(r;\mu t)$ tends to $v_1^{(3)}/(\mu t)$.

When $\beta = \alpha + 1$ the leading term in $f_{\alpha,\beta}(r;\mu t)$ is instead $w_0^{(3)} \log(\mu t(1-r)^2)$.

In the case $\beta > \alpha + 1$ the leading term in $f_{\alpha,\beta}(r;\mu t)$ is $u_0^{(1)}(1-r)^{2(\alpha-\beta+1)}$ for $\beta - \alpha \notin \mathbb{N}$ and $u_0^{(2)}(1-r)^{2(\alpha-\beta+1)}$ for $\beta - \alpha \in \mathbb{N}$. □

When $\beta < \alpha + 1$ the bound for DE_+ does not depend on c and therefore, in order to maximize the strip of analyticity, the value $c = 1$ is used, exactly as in [35].

With $\beta = \alpha + 1$, observe that $\log(\mu t(1-c)^2)$ increases in absolute value for $c \to 1$ but anyway it happens in a very slow way. Therefore in this case we can consider any value $c < 1$ and it is preferable to use c quite close to 1 in order to keep N low (in our experiments we will use $c = 0.92$).

In the following we thus restrict our attention to the most interesting situation $\beta > \alpha + 1$; in this case the error DE_+ of Theorem 1 can be expressed as

$$DE_+ = \mathcal{O}\big((1-c)^{2(\alpha-\beta+1)} e^{-2\pi c/h}\big), \quad h \to 0. \tag{11}$$

As c approaches 1, the term $(1-c)^{2(\alpha-\beta+1)}$ can become extremely large, especially when β is much larger than α.

The error model described in [35] can thus fail when applied to the two–parameters ML function (3), as experimentally observed in Section 2. What we will do in the following is to devise an improved error model which does not disregard the effects of the term $(1-c)^{2(\alpha-\beta+1)}$ on DE_+. Therefore, the optimal parameters will be derived after balancing DE_+, as expressed in (11), with the standard estimates

$$DE_- = \mathcal{O}\big(e^{-\pi^2/(\mu t h^2) + 2\pi/h}\big), \quad TE = \mathcal{O}\big(e^{\mu t(1-(hN)^2)}\big)$$

provided in [35]. With the only aim of simplifying the algebraic derivations we introduce the auxiliary real variable γ such that

$$\frac{\gamma \pi c}{h} = \frac{2\pi c}{h} - 2(\alpha - \beta + 1)\log(1-c); \tag{12}$$

in this way the balancing of the error terms DE_+, DE_- and TE

$$-\frac{\gamma \pi c}{h} = -\frac{\pi^2}{\mu t h^2} + \frac{2\pi}{h} = \mu t\big(1 - (hN)^2\big) \tag{13}$$

leads to the optimal parameters

$$h = \frac{\sqrt{1 + \gamma c(2 + \gamma c)}}{N}, \quad \mu = \frac{\pi N}{(2 + \gamma c) t \sqrt{1 + \gamma c(2 + \gamma c)}} \tag{14}$$

giving an error $E_N = \mathcal{O}\big(e^{-\gamma \pi N c/\sqrt{1+\gamma c(2+\gamma c)}}\big)$ (observe that when $\gamma = 2$ and $c = 1$ the standard parameters (8) are obtained). Thus, an error proportional to a given tolerance $\varepsilon > 0$ is obtained by selecting a number of nodes N given by

$$N = -\frac{\sqrt{1 + \gamma c(2 + \gamma c)}}{\gamma \pi c} \log \varepsilon. \tag{15}$$

For practical use it is preferable to express, thanks to (15), the values of h and μ in terms of the tolerance ε instead of N according to

$$h = -\frac{\gamma \pi c}{\log \varepsilon}, \quad \mu = \frac{-\log \varepsilon}{\gamma c(2 + \gamma c) t}. \tag{16}$$

From (12) and (16) we are able to express the relationship between γ and c as

$$c = c_{\alpha,\beta,\varepsilon}(\gamma) = 1 - \varepsilon^{\frac{\gamma-2}{2(\alpha-\beta+1)\gamma}}. \tag{17}$$

The γ is a free parameter introduced in (12) and the main task now is to choose a suitable value for it. In order to guarantee that $0 < c < 1$, since $\beta > \alpha+1$ and ε can be assumed $0 < \varepsilon < 1$, any value $0 < \gamma < 2$ can be chosen.

It can be convenient to choose $\gamma \in (0,2)$ with the aim of minimizing the computational cost, that is, the nodes number N. At this purpose note that, after replacing the above expression of c in (15) we are able to express N as a function of γ, namely $N = N(\gamma; \varepsilon, \alpha, \beta)$.

The most natural way to choose γ is therefore by minimizing the function $N(\gamma; \varepsilon, \alpha, \beta)$ with respect to γ constrained in $(0,2)$ and we will discuss this aspect later on.

Remark 1 Very often, the case $\alpha = \frac{1}{2}$ and $\beta = 1$ is used in literature for testing the application to ML functions of methods based on the numerical inversion of the Laplace transform. It is arguable that this is a very special case which could provide misleading results if not well–analyzed. Indeed, despite the fact that $\beta > \alpha$, and hence a singularity appears at the origin, one obtains excellent results, usually better than those expected, without having to worry about the location of the singularity (this phenomena is clearly observable in Figure 1). Based on the above analysis it is possible to explain this favorable situation. Indeed, since it is $\Psi(\frac{1}{2}, \frac{3}{2}, z) = 1/\sqrt{z}$ (see [36]), the bound

$$\int_{-\infty}^{\infty} \bigl|g(u+ir)\bigr| du \le \frac{M}{\sqrt{\mu t}}$$

holds not only asymptotically (for $r \to 1$), as it in general happens in Theorem 3, but for any value of r. For this reason an excellent performance is achieved regardless of the choice of a threshold c for the strip of analyticity when $\alpha = \frac{1}{2}$ and $\beta = 1$.

3.2 The case $1 < \alpha < 2$

When $1 < \alpha < 2$ a more involved analysis is necessary because of the presence of the additional singularities $s^{-}_{\alpha,\lambda} = \lambda^{\frac{1}{\alpha}} e^{-\pi/\alpha}$ and $s^{+}_{\alpha,\lambda} = \lambda^{\frac{1}{\alpha}} e^{\pi/\alpha}$ (actually, the symmetry of the parabolic contour (7) with respect to the real axis slightly simplifies our analysis since only one of these two singularities, say $s^{+}_{\alpha,\lambda}$, needs to be taken into account).

Two main goals need to be added in the procedure for the parameters selection: 1) constraining the parabola to encompass the singularity $s^{+}_{\alpha,\lambda}$; 2) taking into account $s^{+}_{\alpha,\lambda}$ in determining the width c of the strip of analyticity for the estimation of the error in Theorem 1.

For the first goal, it is elementary, after equating $z(u) = s^{+}_{\alpha,\lambda}$, to show that the parabola intersects $s^{+}_{\alpha,\lambda}$ when $\mu = \lambda^{\frac{1}{\alpha}} \bigl(\cos\frac{\pi}{\alpha} + 1\bigr)/2$ and hence, to assure that the parabola stays in the complex plane at the right of the singularities, it is necessary to require that

$$\mu > \bar{\mu} = \frac{\lambda^{\frac{1}{\alpha}}}{2}\bigl(\cos\tfrac{\pi}{\alpha} + 1\bigr). \tag{18}$$

For the second goal we have to find the maximum value c such that the parabola given by $z(u+ir)$ does not touch the singularity $s_{\alpha,\lambda}^{+}$ for any $0 < r < c$. Also in this case it is easy to see that, after equating $z(u+ir) = s_{\alpha,\lambda}^{+}$, we can determine

$$c < 1 - \sqrt{\frac{\lambda^{\frac{1}{\alpha}}}{2\mu}\bigl(\cos\tfrac{\pi}{\alpha} + 1\bigr)}. \tag{19}$$

Long and complicated calculations allow to translate this condition into a minimization problem with nonlinear functional constraints. The solution of a problem of this kind is very demanding in a method which aims to be simple and elegant. As we will observe later, it is not necessary to solve exactly the minimization problem, since a perturbation with respect to the minimum point will increase the number of grid–points of only few unities. Thus, to reduce the complexity we propose an alternative procedure which, although not optimal, provides satisfactory enough results as we will see later by numerical experiments.

A first value $\bar{\gamma}$ is obtained by minimizing the function $N(\gamma; \varepsilon, \alpha, \beta)$ (when $\beta \leq \alpha + 1$ it is simply $\bar{\gamma} = 2$) and a corresponding value $\bar{c}$ is obtained thanks to (17). Hence, using (16) the test (18) becomes

$$\bar{\gamma}^2 t c^2 + 2\bar{\gamma} t c + \frac{\log \varepsilon}{\bar{\mu}} < 0$$

and, if it is not fulfilled, a value c satisfying the above inequality, i.e.

$$c < \frac{1}{\bar{\gamma}}\left(\sqrt{1 - \frac{\log\varepsilon}{\bar{\mu} t}} - 1\right)$$

will replace the previous value of $\bar{c}$. This new value is accepted if it satisfies (19), i.e.

$$(1 - \varphi\bar{\gamma}^2 t)c^2 - 2(1 + \varphi\bar{\gamma} t)c + 1 > 0,$$

where for shortness we put $\varphi = -\lambda^{\frac{1}{\alpha}}\bigl(\cos\frac{\pi}{\alpha} + 1\bigr)/(2\log\varepsilon)$, otherwise $\bar{c}$ is again replaced in a proper way. Finally, in order to preserve the same product $\bar{\gamma}\bar{c}$, also γ is updated as $\gamma = \bar{\gamma}\bar{c}/c$.

4 Numerical experiments

To validate the results discussed, from a theoretical point of view, in the previous sections, we present some numerical experiments.

After selecting a given accuracy ε that we aim to reach, we perform the computation of the generalized ML function (3) according to the procedure described in Section 3 and we show the resulting error and the number N of grid points (this last information provides an idea on the computational cost of the algorithm).

As reference solution we use the results provided by the Matlab `mlf` code [27]. This is a very accurate and robust code allowing to evaluate the classical ML function in any region of the complex plane at any prescribed tolerance; anyway it is not always useful for practical computation since it works very slowly for relatively small values of α and does not allow to evaluate the function for matrix arguments. All the experiments are performed by using Matlab ver. 7.7.

The results in Tables 1 and 2 show that the procedure succeeds in reaching the requested tolerance $\varepsilon = 10^{-8}$ with a number of grid–points rather small. Similar results are provided in Tables 3 and 4 where a target accuracy of $\varepsilon = 10^{-11}$ has been requested.

α	β	Error	N	α	β	Error	N	α	β	Error	N
0.2	0.2	4.95(−9)	9	0.5	0.5	2.63(−9)	9	0.8	0.8	1.09(−9)	9
0.2	1.0	1.49(−9)	9	0.5	1.0	4.24(−10)	9	0.8	1.0	4.98(−10)	9
0.2	1.2	2.06(−10)	10	0.5	1.5	2.11(−10)	10	0.8	1.8	2.03(−10)	10
0.2	2.2	4.12(−10)	10	0.5	2.5	4.63(−10)	10	0.8	2.8	4.84(−10)	10
0.2	3.2	2.82(−11)	11	0.5	3.5	3.16(−11)	11	0.8	3.8	3.37(−11)	11
0.2	4.2	1.24(−12)	12	0.5	4.5	1.37(−12)	12	0.8	4.8	1.47(−12)	12

Table 1 Errors for $\lambda = 2.5$ and $t = 1.0$ after requiring $\varepsilon = 10^{-8}$.

α	β	Error	N	α	β	Error	N	α	β	Error	N
1.2	1.0	8.72(−10)	10	1.5	1.0	8.01(−10)	11	1.8	1.0	2.09(−9)	11
1.2	1.2	7.98(−10)	10	1.5	1.5	6.38(−11)	11	1.8	1.8	1.69(−10)	11
1.2	2.2	1.61(−10)	10	1.5	2.5	2.20(−10)	11	1.8	2.8	1.03(−9)	11
1.2	3.2	4.89(−10)	10	1.5	3.5	6.05(−9)	10	1.8	3.8	7.48(−8)	10
1.2	4.2	3.49(−11)	11	1.5	4.5	3.34(−11)	11	1.8	4.8	3.10(−10)	11
1.2	5.2	1.54(−12)	12	1.5	5.5	1.47(−12)	12	1.8	5.8	1.18(−12)	12

Table 2 Errors for $\lambda = 2.5$ and $t = 1.0$ after requiring $\varepsilon = 10^{-8}$.

We now present some results concerning the matrix case. Our main goal is to confirm the high level accuracy one can reach by approximating the generalized ML function by means of the integration on the parabolic contour with optimally chosen parameters. We consider the problem of approximating $e_{\alpha,\beta}(t;A)v$ for a square matrix A and a vector v since, as discussed in the Introduction, this is a very common problem in applications.

We consider three test matrices: the first one comes from the discretization of the Laplace operator in 2 dimensions by means of centered finite differences;

α	β	Error	N	α	β	Error	N	α	β	Error	N
0.2	0.2	1.32(−12)	13	0.5	0.5	6.34(−13)	13	0.8	0.8	2.36(−13)	13
0.2	1.0	3.15(−13)	13	0.5	1.0	8.59(−14)	13	0.8	1.0	1.00(−13)	13
0.2	1.2	2.87(−13)	13	0.5	1.5	2.82(−13)	13	0.8	1.8	2.69(−13)	13
0.2	2.2	7.51(−14)	14	0.5	2.5	8.42(−14)	14	0.8	2.8	8.78(−14)	14
0.2	3.2	4.75(−15)	15	0.5	3.5	4.66(−15)	15	0.8	3.8	5.73(−15)	15
0.2	4.2	3.00(−15)	15	0.5	4.5	3.47(−15)	15	0.8	4.8	3.55(−15)	15

Table 3 Errors for $\lambda = 2.5$ and $t = 1.0$ after requiring $\varepsilon = 10^{-11}$.

α	β	Error	N	α	β	Error	N	α	β	Error	N
1.2	1.0	4.32(−14)	14	1.5	1.0	1.13(−12)	14	1.8	1.0	7.25(−13)	15
1.2	1.2	8.40(−14)	14	1.5	1.5	1.62(−12)	14	1.8	1.8	4.02(−13)	15
1.2	2.2	3.24(−14)	14	1.5	2.5	2.11(−13)	14	1.8	2.8	2.06(−13)	15
1.2	3.2	9.11(−14)	14	1.5	3.5	2.40(−12)	14	1.8	3.8	7.31(−11)	14
1.2	4.2	5.66(−15)	15	1.5	4.5	1.11(−14)	15	1.8	4.8	5.89(−14)	15
1.2	5.2	3.70(−15)	15	1.5	5.5	4.01(−15)	15	1.8	5.8	5.98(−15)	15

Table 4 Errors for $\lambda = 2.5$ and $t = 1.0$ after requiring $\varepsilon = 10^{-11}$.

so it is a symmetric positive definite matrix. The second matrix we consider comes from the discretization of the differential operator

$$D = -\frac{\partial^2}{\partial x^2} - \frac{\partial^2}{\partial y^2} + 10x\frac{\partial}{\partial x},$$

on the square $(0,1) \times (0,1)$, with homogeneous Dirichlet conditions. The third matrix comes from the Matlab matrix collection and is the `chebspec` operator. For the second case the vector v is taken as the (normalized) discretization of the function $v(x,y) = x(1-x)y(1-y)$, while in the other cases v is taken as the normalized unit vector.

The first matrix is symmetric while the latter two are examples of non symmetric matrices with real eigenvalues. We stress that the hypothesis on the real nature of the spectrum is not always sufficient to ensure good results, since high non–normality severely deteriorates the error analysis, as deeply discussed in [14,34]. However we do not deepen this topic as the matrices we consider do not fall into this case.

We compare the error of our approach with respect to a reference solution obtained by combining the Matlab `funm` function and the `mlf` function [27] carried out with a very high accuracy. In all cases the matrix dimension is 400: this dimension is small enough to solve the systems with a direct method but, at the same time, it is large enough to start feeling the difficulties related to the Schur factorization in the `funm` method. We consider $\alpha = 0.2, 0.8$ and $\beta = \alpha, \alpha+1, \alpha+2, \alpha+3$, for $t = 1$. The required accuracy is $\varepsilon = 10^{-8}$. Similar results have been obtained for several other values of α, of the tolerance and of the matrix dimensions; for this reason we omitted their presentation.

α	β	Matrix 1	Matrix 2	Matrix 3	N
0.2	0.2	5.45(−10)	9.23(−11)	3.02(−10)	9
	1.2	8.03(−11)	2.15(−10)	1.28(−11)	9
	2.2	1.17(−10)	2.16(−10)	2.57(−11)	10
	3.2	6.94(−12)	1.06(−10)	1.80(−12)	11
0.8	0.8	1.11(−10)	5.04(−11)	6.03(−11)	9
	1.8	1.04(−09)	2.17(−10)	1.28(−11)	10
	2.8	5.98(−10)	2.16(−10)	3.04(−11)	10
	3.8	2.27(−11)	1.05(−10)	2.12(−12)	11

Table 5 Errors for the matrix examples of dimension 400 after requiring $\varepsilon = 10^{-8}$; $t = 1.0$.

Results in Table 5 confirm those obtained for the scalar case but they emphasize the smartness of the approach: it is indeed striking that by simply solving a dozen systems a very high accuracy is reached.

In analogy with the scalar case, we notice that in spite of the required accuracy 10^{-8} we often reach a higher one. However this does not represent a serious issue since this extra accuracy comes with a value N which is very small.

As in the scalar case we notice that the number N increases as β gets larger than α.

These results are very promising and they are encouraging also for applications of much larger dimensions. In this case the only difference would be the solver for the linear systems, since iterative techniques would be probably mandatory. Moreover, the use of preconditioning could help in the case of slow convergence, even though this subject for complex systems is still an active research area and we just cite [1,2].

5 Computational efficiency

In this section we discuss some issues related to the efficiency of the procedure investigated in this paper.

5.1 Minimization of the function $N(\gamma; \varepsilon, \alpha, \beta)$

The task of minimizing the function $N(\gamma; \varepsilon, \alpha, \beta)$, under the constraint $0 < \gamma < 2$, may appear to be particularly challenging. In our experiments we have performed it by means of the Matlab function `fminbnd` which uses an algorithm based on the golden section search with parabolic interpolation.

As shown in Figure 2, the function $N(\gamma; \varepsilon, \alpha, \beta)$ features a flat slope in the neighborhood of its minimum; therefore, it is not necessary to locate the point of minimum with an extremely high accuracy. We have experienced that satisfactory results are obtained by using a relatively small tolerance of 10^{-4} in `fminbnd`; this choice involves less than a dozen of evaluations of $N(\gamma; \varepsilon, \alpha, \beta)$ and the whole process turns out to be computationally cheap.

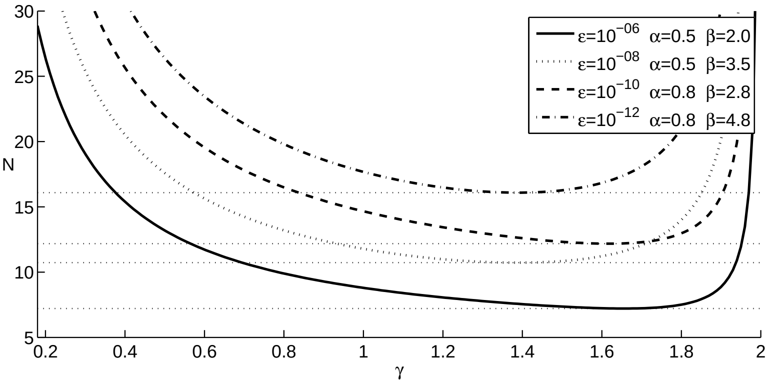

Fig. 2 Behavior of $N(\gamma;\varepsilon,\alpha,\beta)$ with respect to γ for some values of ε, α and β.

Frequently the function $e_{\alpha,\beta}(t;\lambda)$ is evaluated with $\beta=\alpha+k$ for few $k\in\mathbb{N}$ and several values of t, as discussed in the next subsection. Since in this case the function $N(\gamma;\varepsilon,\alpha,\beta)$ does not depend on α,β and t but just on k, the point of minimum could be evaluated once for several couples (ε,k) and the resulting values N tabled and reused for each value of t without repeatedly performing the minimization of $N(\gamma;\varepsilon,\alpha,\beta)$.

In the end, the evaluation of the minimum of $N(\gamma;\varepsilon,\alpha,\beta)$ does not seem to represent an overshadowing portion of the whole computation.

5.2 Computation of $e_{\alpha,\beta}(t;\lambda)$ for $t\in[t_0,t_1]$

Important applications require the simultaneous computation of $e_{\alpha,\beta}(t;\lambda)$ for several t's all belonging to a given interval $[t_0,t_1]$. This requirement is common, for example, when dealing with the GETD methods [5] and it actually represents a crucial point of these techniques. The procedure described above, based on a parabola, is strictly related to the value t. However, if we look at the error terms in (13), we see that the term t appears only in DE_- and TE. More precisely, DE_- decreases when t gets large, while TE has the opposite behavior. Thus, to maintain all error terms small in $[t_0,t_1]$, by following the ideas in [35], we could replace (13) with

$$2(\alpha-\beta+1)\log(1-c)-\frac{2\pi c}{h}=-\frac{\pi^2}{\mu t_1 h^2}+\frac{2\pi}{h}=\mu t_0\bigl(1-(hN)^2\bigr).$$

The analysis for the parabola parameters can be done as before, leading to

$$h=\frac{\sqrt{1+\gamma\bar{c}R(2+\gamma\bar{c})}}{N},\quad \mu=\frac{\pi N}{(2+\gamma\bar{c})t_1\sqrt{1+\gamma\bar{c}R(2+\gamma\bar{c})}}$$

for $R=t_1/t_0$, which trivially reduces to the previous case for a single t, by choosing $t=t_0=t_1$.

5.3 Optimality of the error model

As shown by the numerical experiments, the error model described in Section 3 succeeds to evaluate our generalized ML function with an accuracy under any fixed tolerance. Anyway, from the results presented one could get the feeling that the procedure is not altogether optimal; indeed, the errors actually obtained are sometimes smaller than requested; usually two or three digits of extra precision are provided. Obviously this does not constitute a problem for the reliability of the results; anyway, the extra precision could mean that the algorithm determines a number N of grid–points greater than strictly necessary.

We can therefore argue the existence of a set of parameters allowing to reach the target accuracy with a smaller number of grid–points and hence saving some computational cost. To closely examine this aspect we observe Figure 3 where the error $\log_{10}\bigl|e_{\alpha,\beta}(t;\lambda) - e_{\alpha,\beta}^{[N]}(t;\lambda)\bigr|$ obtained for a large set of parameters h and λ and with a fixed $N = 12$ has been reported; here the values $\alpha = 0.5$, $\beta = 4.5$, $t = 1.0$ and $\lambda = 1.5$ have been used. The black cross locates the couple of parameters (μ, h) as determined in [35] by (8). As already observed, this selection of parameters provides a poor quality approximation when β is fairly larger than α as in the example under examination. The white line denotes the set of (μ, h) obtained from (14) by moving the product γc in the interval $[0, 2]$; obviously, when $\gamma c = 2$, h and μ are the same parameters estimated by (8); thus, the white line starts just in the position of the black cross. As the product γc decreases from the value 2, the corresponding parameters h and μ generate errors of smaller size. The path of this white line is very close to the area with the smallest errors, thus indicating that the selection given by (14) is in the right direction for obtaining the finest results and validating our error model. Anyway, the procedure described in Section 3 selects a small range of values indicated by the gray segment laying upon the white line (to obtain this range we have used the procedure described in Section 3 for several values of ε and selected the entries giving the value $N = 12$).

As expected, our procedure improves the results in [35] for the generalized ML function but does not select the best set of parameters in order to minimize the error with a given number of grid–points: from Figure 3 we deduce that it is possible to obtain a smaller error, with the same number $N = 12$ of grid–points if only we were able to select some other couple (h, μ) of the parameters by moving on the same white line.

The source for this slight inaccuracy is probably the fact that the error estimates are asymptotic and hold as $N \to \infty$. Indeed, we have reproduced the analysis of Figure 3 for larger values of N and we found results more adherent to the theoretical findings. We omitted to present these results, as the plots corresponding to larger values of N are harder to be read since the errors become very close to the machine precision.

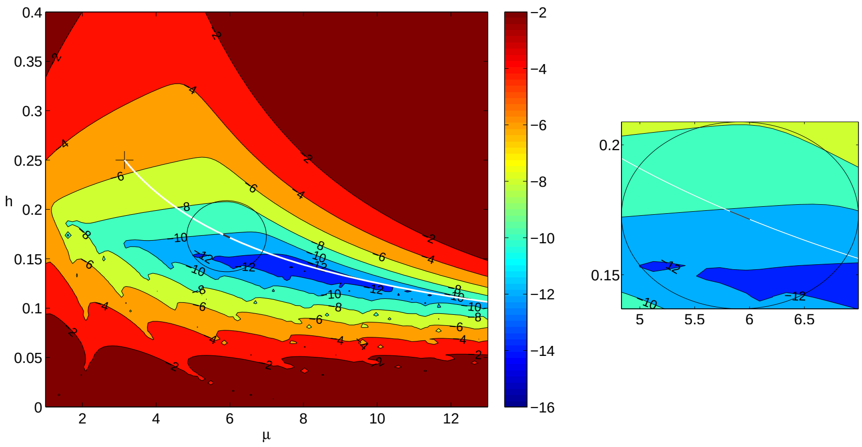

Fig. 3 Errors $\log_{10}\left|e_{\alpha,\beta}(t;\lambda) - e^{[N]}_{\alpha,\beta}(t;\lambda)\right|$ for $\lambda = 1.5$, $t = 1.0$ and $N = 12$.

6 Concluding remarks

The results presented in this paper confirm the effectiveness of the approach described in [35] when employed for evaluating the generalized ML function (3). Moreover, the analysis performed in Section 3, validated by numerical experiments, allows to tune in a fine way the parameters of the method in order to evaluate $e_{\alpha,\beta}(t;\lambda)$ under any target accuracy for any value of α and β.

Due to its modest computational cost the method appears to be suitable also for the computation of the $e_{\alpha,\beta}$ function with matrix arguments, as in the generalization of exponential integrators to FDEs, a topic which is recently gaining an increasing attention.

Although parabolic contours are particularly attracting for the simplicity of their geometry and the elegance of the involved analysis, the sectorial nature of the inverse Laplace transform of the Mittag–Leffler function (as stated in Proposition 1) could be better exploited by integrating on contours of hyperbolic type, thus allowing the approximation of the ML function also for more general operators. We think that this issue deserves a deeper investigation which is beyond the scope of the present paper and will be the subject for future research.

Acknowledgements The authors are grateful to the anonymous referees for their helpful and constructive comments on an earlier version of the paper.

The work of R. Garrappa was supported by the Italian Ministry of University and Research (MIUR), project grant n. 2009F4NZJP. The work of M. Popolizio was supported by the GNCS–INdAM 2012 project “Integratori esponenziali per equazioni differenziali di ordine frazionario”.

References

1. Benzi, M., Bertaccini, D.: Real-valued iterative algorithms for complex symmetric linear systems. IMA J. Numer. Anal. **28**(3), 598–618 (2008)
2. Bertaccini, D.: Efficient preconditioning for sequences of parametric complex symmetric linear systems. Electron. Trans. Numer. Anal. **18**, 49–64 (electronic) (2004)
3. Diethelm, K., Ford, N.J., Freed, A.D., Luchko, Y.: Algorithms for the fractional calculus: a selection of numerical methods. Computer Methods in Applied Mechanics and Engineering **194**(6), 743–773 (2005)
4. Diethelm, K.: The analysis of fractional differential equations, *Lecture Notes in Mathematics*, vol. 2004. Springer-Verlag, Berlin (2010)
5. Garrappa, R., Popolizio, M.: Generalized exponential time differencing methods for fractional order problems. Computers and Mathematics with Applications **62**(3), 876–890 (2011)
6. Garrappa, R., Popolizio, M.: On accurate product integration rules for linear fractional differential equations. J. Comput. Appl. Math. **235**(5), 1085–1097 (2011)
7. Garrappa, R., Popolizio, M.: On the use of matrix functions for fractional partial differential equations. Math. Comput. Simulation **81**(5), 1045–1056 (2011)
8. Garrappa, R.: Stability-Preserving High-Order Methods for Multiterm fractional differential equations. International Journal of Bifurcation and Chaos, **22**(4), (2012)
9. Gil, A., Segura, J., Temme, N.M.: Numerical methods for special functions. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA (2007)
10. Gorenflo, R., Loutchko, J., Luchko, Y.: Computation of the Mittag-Leffler function and its derivatives. Fractional Calculus and Applied Analysis **5**(4), 491–518 (2002)
11. Gorenflo, R., Mainardi, F.: Fractional oscillations and Mittag-Leffler functions. Tech. Rep. A-14/96, Freie Universitaet Berlin, Serie A Mathematik (1996)
12. Haubold, H.J., Mathai, A.M., Saxena, R.K.: Mittag–Leffler functions and their applications. Journal of Applied Mathematics pp. Art. ID 298,628, 51 (2011)
13. Hilfer, R. Seybold, H.J.: Computation of the generalized Mittag-Leffler function and its inverse in the complex plane. Integral Transforms Spec. Funct. **17**(9), 637–652 (2006)
14. in 't Hout, K. J., Weideman, J.A.C.: A contour integral method for the Black–Scholes and Heston equations. SIAM Journal on Scientific Computing. **33**(2), 763–785, (2011)
15. Kilbas, A.A., Srivastava, H.M., Trujillo, J.J.: Theory and applications of fractional differential equations, *North-Holland Mathematics Studies*, vol. 204. Elsevier Science B.V., Amsterdam (2006)
16. López-Fernández, M., Palencia, C.: On the numerical inversion of the Laplace transform of certain holomorphic mappings. Appl. Numer. Math. **51**(2-3), 289–303 (2004)
17. López-Fernández, M., Palencia, C., Schädle, A.: A spectral order method for inverting sectorial Laplace transforms. SIAM J. Numer. Anal. **44**(3), 1332–1350 (electronic) (2006)
18. Lorenzo, C.F., Hartley, T.T.: Generalized functions for the fractional calculus. Crit. Rev. Biomed. Eng. **36**(2), 39–55 (2008)
19. Mainardi, F.: Fractional calculus and waves in linear viscoelasticity. Imperial College Press, London (2010). DOI 10.1142/9781848163300
20. Mainardi, F., Gorenflo, R.: On Mittag-Leffler-type functions in fractional evolution processes. J. Comput. Appl. Math. **118**(1-2), 283–299 (2000)
21. Miller, K.S., Ross, B.: An introduction to the fractional calculus and fractional differential equations. A Wiley-Interscience Publication. John Wiley & Sons Inc., New York (1993)
22. Miller, K.S., Samko, S.G.: A note on the complete monotonicity of the generalized Mittag-Leffler function. Real Anal. Exchange **23**(2), 753–755 (1997/98)
23. Moret, I., Novati, P.: On the convergence of Krylov subspace methods for matrix Mittag–Leffler functions. SIAM J. Numer. Anal. **49**(5), 2144–2164 (2011)
24. Moret, I., Popolizio, M.: The restarted shift–invert Krylov method for matrix functions (2011). Submitted
25. Oldham, K.B., Spanier, J.: The fractional calculus. Academic Press (1974)
26. Podlubny, I.: Fractional differential equations, *Mathematics in Science and Engineering*, vol. 198. Academic Press Inc., San Diego, CA (1999)

27. Podlubny, I., Kacenak, M.: The Matlab mlf code. MATLAB Central File Exchange (2001–2009). File ID: 8738
28. Saxena, R.K., Saigo, M.: Certain properties of fractional calculus operators associated with generalized Mittag-Leffler function. Fract. Calc. Appl. Anal. **8**(2), 141–154 (2005)
29. Seybold, H., Hilfer, R.: Numerical algorithm for calculating the generalized Mittag-Leffler function, SIAM J. Numer. Anal. **47**(1), 69–88 (2008/09)
30. Trefethen, L.N., Weideman, J.A.C., Schmelzer, T.: Talbot quadratures and rational approximations. BIT **46**(3), 653–670 (2006)
31. Tricomi, F.G.: Funzioni ipergeometriche confluenti. Edizione Cremonese, Roma (1954)
32. Verotta, D.: Fractional compartmental models and multi–term Mittag-Leffler response functions. J. Pharmacokinet Pharmacodyn. **37**(2), 209–215 (2010)
33. Weideman, J.A.C.: Computing special functions via inverse Laplace transforms. In: T. Simos, G. Psihoyios, C. Tsitouras (eds.) International Conference on Numerical Analysis and Applied Mathematics 2005 (Rhodes), 1, pp. 702–704. Wiley-VCH (2005)
34. Weideman, J.A.C.: Improved contour integral methods for parabolic PDEs. IMA J. Numer. Anal. **30**(1), 334–350 (2010)
35. Weideman, J.A.C., Trefethen, L.N.: Parabolic and hyperbolic contours for computing the Bromwich integral. Math. Comp. **76**(259), 1341–1356 (electronic) (2007)
36. Wolfram Research Inc.: Tricomi confluent hypergeometric function (1998–2010). URL http://functions.wolfram.com